Research Article

PRO: A Novel Approach to Precision and Reliability Optimization Based Dominant Point Detection

Dilip K. Prasad

School of Computing, National University of Singapore, Singapore 117417

Correspondence should be addressed to Dilip K. Prasad; dilipprasad@gmail.com

Received 5 June 2013; Revised 21 July 2013; Accepted 2 August 2013

Academic Editor: Manuel Lozano

Copyright © 2013 Dilip K. Prasad. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

A novel method that uses both the local and the global nature of fit for dominant point detection is proposed. Most other methods use local fit to detect dominant points. The proposed method uses simple metrics like precision (local nature of fit) and reliability (global nature of fit) as the optimization goals for detecting the dominant points. Depending on the desired level of fitting (very fine or crude), the threshold for precision and reliability can be chosen in a very simple manner. Extensive comparison of various line fitting algorithms based on metrics such as precision, reliability, figure of merit, integral square error, and dimensionality reduction is benchmarked on publicly available and widely used datasets (Caltech 101, Caltech 256, and Pascal (2007, 2008, 2009, 2010) datasets) comprising 102628 images. Such work is especially useful for segmentation, shape representation, activity recognition, and robust edge feature extraction in object detection and recognition problems.

1. Introduction

In many applications, boundaries are represented using polygonal approximation [1–8]. The problem of dominant points detection is to determine the points only from a digital curve for such representation. This representation reduces the memory and computational complexity in storing and processing the digital curves and helps in the determination of geometrical properties like inflexion points, perimeter, and tangent estimation. It is useful for topological representation, character recognition, segmentation, and contour feature extraction in the applications of computer vision. Further, it reduces the problems of digitization and related noise issues.

The problem of fitting lines on curves (including dominant point detection) is quite old. The method of Teh and Chin [9] relies primarily on the accurate determination of the support region based on chord length and the perpendicular distance of the pixels from the chords to determine the dominant points. Ansari and Huang [10] proposed a method in which a support region is assigned to each boundary point based on its local properties. A combination of Gaussian filtering and a significance measure is used on each pixel for identifying the dominant points. Cronin's [11] method

finds the support region for every pixel based on a non-uniform significance measure criterion calculated by locally determining the support region for each point. B. K. Ray and K. S. Ray [12] proposed a k-cosine-transform based method to determine the support region. Sarkar [13] proposed purely a chain code manipulation based method for determining the dominant points where the chain code is sufficient and the exact coordinates of the pixels are not necessary.

This problem remains relevant in the current era as well. Some of the recent dominant point detection methods are Masood [14] and its modification [15], Carmona-Poyato et al. [16], and Nguyen and Debled-Rennesson [17]. These methods have already shown considerable improvements over earlier dominant point detection or polygonal approximation methods. However, all of these methods except that of Carmona-Poyato et al. [16] use local properties of fit like the maximum distance (deviation) of the pixels on the digital curve from the fitted polygon. On the other hand Carmona-Poyato et al. [16] use a ratio “ r ” which incorporates the quality of the global fit instead of the local fit.

We highlight the reasons for the continued relevance of this problem in the current era. Even though many algorithms were developed, the measures or metrics used

to compare and benchmark various algorithms were not effective, as shown in [18, 19]. Researchers tried using absolute measures such as compression ratio, integral square error, figure of merit, zero norm, and infinity norm; many studies proved that such metrics fail to represent the quality of fit in one manner or other. The reason for this was not understood fully. Recently, using very simple metrics, namely, precision and reliability measures, it was shown that there is a perennial conflict in the quality of fit in the local scale (precision at the level of a few pixels) and global scale (reliability at the level of complete curve) [20–22]. The duality of precision/reliability of a fit haunts most fitting algorithms. The reliability of the fitting decreases with the increase in the precision of fitting and vice versa [20, 21, 23]. Due to this reason, most absolute measures fail in quantifying the quality of fit properly. However, due to the conflicting nature of the global and local natures of fit, it is better to optimize both of them simultaneously. The present work is a first attempt in our knowledge towards this aim. In this paper, we present three contributions.

- (i) The proposed algorithm provides the user with good flexibility regarding the nature of fit. For example, the user may choose a very close fit, which closely follows the digital curve and retains all the small perturbations in the curve. Or the user may choose a curvature following fit that removes the local effects (due to noise, etc.) and retains the large-scale features of the digital curve.
- (ii) The precision and reliability measures are extended for digital curves. In [20], these measures were only defined for straight line segments. Now, these are now extended as the measures for fitting line segments on digital curves.
- (iii) An important contribution of the proposed work is the numerical experiments. A total of 102628 images from 10 datasets are used for analysis [24–32]. These datasets are chosen because they are the benchmark datasets in the field of computer vision and the research and applications using these datasets often employ dominant point detection as a fundamental preprocessing step.

We highlight that the focus of this paper is to present a method that gives the user the flexibility to choose the nature of fit very easily using an intuitive and simple control parameter. References [33, 34] are examples of practical problems that need such flexibility. Thus the aim of this paper is quite different from [35–37], where the aim was to develop a framework (not a method) to make standard methods nonparametric and stable within the digitization errors.

2. The Basic Framework of Line Fitting

The basic framework is similar to the dominant point detection method proposed by Lowe [38] and Ramer-Douglas-Peucker [39, 40] (referred to as the RDP method). We highlight that the methods of Lowe [38] and Ramer-Douglas-Peucker [39, 40] are not exactly the same. However, both

are based on splitting at the point of maximum deviation. Lowe [38] has an additional merge stage and uses a curvature parameter which makes it relatively less control parameter dependent. Here, the main concept and splitting stage are of relevance. Let us consider a digital curve $e = \{P_1 P_2 \cdots P_N\}$, where P_i is the i th edge pixel in the digital curve e . The line passing through a pair of pixels $P_a(x_a, y_a)$ and $P_b(x_b, y_b)$ is given by (1)

$$x(y_a - y_b) + y(x_b - x_a) + y_b x_a - y_a x_b = 0. \quad (1)$$

Then the deviation d_i of a pixel $P_i(x_i, y_i) \in e$ from the line passing through the pair $\{P_1, P_N\}$ is given as (2)

$$d_i = \frac{|x_i(y_1 - y_N) + y_i(x_N - x_1) + y_N x_1 - y_1 x_N|}{\sqrt{(x_N - x_1)^2 + (y_1 - y_N)^2}}. \quad (2)$$

Accordingly, the pixel with maximum deviation can be found. Let it be denoted as $P_{\max}$. Then considering the pairs $\{P_1, P_{\max}\}$ and $\{P_{\max}, P_N\}$, we find two new pixels from e using the concept expressed in (1) and (2). It is evident that the maximum deviation decreases as one chooses newer pixels of maximum deviation between a pair. This process can be repeated till a certain condition (depending upon the method) is satisfied by all the line segments. This condition will be referred to as the optimization goal for the ease of reference. The condition used by RDP [38–40] is that for each line segment, the maximum deviation of the pixels contained in its corresponding edge segment is less than a certain tolerance value:

$$\max(d_i) < d_{\text{tol}}, \quad (3)$$

where d_{tol} is the chosen threshold and is typically a few pixels.

3. Precision- and Reliability-Based Optimization (PRO)

The precision and reliability measures were defined for a line segment in [20]. Suppose, for a sequence of connected pixels $S = \{P_i(x_i, y_i), i = 1 \text{ to } M\}$, a line $ax + by = 1$ fits the pixels perfectly. Then, the coefficients of the line, a and b , must satisfy (4)

$$\mathbf{X}\bar{\mathbf{A}} = \bar{\mathbf{J}}, \quad (4)$$

where $\mathbf{X} = [[x_1 \ x_2 \ \cdots \ x_M]^T [y_1 \ y_2 \ \cdots \ y_M]^T]$, $\bar{\mathbf{A}} = [a \ b]^T$, the superscript T denotes the transpose operation, and $\bar{\mathbf{J}}$ is a column matrix containing M rows, whose every element is 1. Then, using (4) as the ideal case, the precision of fitting can be modeled using the normalized residue given in (5)

$$\epsilon'_p = \frac{|\bar{\mathbf{J}} - \mathbf{X}\bar{\mathbf{A}}|}{|\bar{\mathbf{A}}|}, \quad (5)$$

where $|\cdot|$ represents the Euclidean norm of vectors. Since it considers the residue for each pixel, it is characteristic

```

Function DP = PRO ({P1, P2, ..., PN})
{
  DP = NULL; % DP contains the dominant points
  % Step 1. line and its parameters
  Fit a line l using P1 and PN.
  % Step 2. maximum deviation
  Find deviation {d1, ..., dN} of pixels {P1, ..., PN} from the line l.
  Find dmax = max{d1, ..., dN} and point Pmax corresponding to dmax.
  Find ε'p and εr.
  % Step 3. termination/recursion condition
  If (ε'p ≤ ε0) AND (εr ≤ ε0)
    DP = {DP, P1, PN}
  Else
    { DP = {DP, PRO (P1, Pmax)}.
    DP = {DP, PRO(Pmax, PN)}.
    }
  End
  Remove redundant points in DP.
  Return(DP).
}

```

ALGORITHM 1: Pseudocode for PRO algorithm.

of the local nature of fitting alone. We highlight that the precision metric is closely related to conventionally used integral square error (see (16)). However, the present form in (5) makes it comparable with the reliability metric presented next, such that the same control parameter can be used for both precision and reliability metrics.

On the other hand, for the global characteristics of fit, another measure called the reliability measure is needed. Generally, reliability of a fit refers to how well the fit is expected to satisfy at least two conditions:

- (i) the fit should be valid for a sufficiently large region (or in this case a long curve),
- (ii) it should not be sensitive to occasional spurious large deviations in the edge.

A combination of both these properties can be sought by defining a reliability measure as shown in (6)

$$\varepsilon_r = \frac{\sum_i |ax_i + by_i - 1|}{s_{\max}}, \quad (6)$$

where $s_{\max}$ is the maximum Euclidean distance between any two pair of pixels [20]. Here, $|\cdot|$ represents the magnitude or the absolute value.

In RDP, when a point on the curve is selected as a dominant point, the curve is segmented into two smaller curves and two line segments intersecting at the selected dominant point are used to represent the original curve. The two line segments are associated directly with the two smaller curves. Thus, for each pair of line segment and its associated curve, one can compute the precision and reliability measures ε'_p and ε_r using (5) and (6). Then, for each segmented curve, if the precision and reliability measures are not below a certain threshold given by (7)

$$\max(\varepsilon'_p, \varepsilon_r) < \varepsilon_0, \quad (7)$$

then the segmented curve is further segmented into two curves at the point of maximum deviation. Then two independent lines are fitted on these two subcurves using the least squares method and the whole procedure is performed recursively until the reliability and precision measures for each line fit on each subcurve are lesser than the chosen threshold. Typically, ε_0 is the chosen tolerance value and is typically less than 1. The pseudocode for the PRO algorithm is provided in Algorithm 1.

In the proposed modification the same algorithm as RDP is used to fit the curves, but the optimization goal is not based on maximum deviation (use of (3) as the optimization goal). It is based on precision and reliability measures ε'_p and ε_r (use of (7) as the optimization goal), which give the users larger freedom in determining the nature of fit. For example, using ε_0 close to 0 (like 0.1 or 0.2) results in a very close fit on the curve, where the lines follow even small deviations in the curvature. This may be used to achieve high fidelity fit, where one needs to retain even the smallest deviations in the data or where one studies the nature of noise in the curve itself. Using ε_0 close to 1 (like 0.9 or 1) results in line fit that smoothens over small spurious deviations and retains all significant curvature changes in the line fit.

4. Precision and Reliability Measures for Fitting Line Segments on Digital Curves

This section extends the precision and reliability measures expressed in (5) and (6) to the cases of digital curves. Suppose J line segments are fitted upon a digital curve. Then the net precision measure for the digital curve is defined as follows (8) [21]:

$$\varepsilon'_p = \text{mean}(\varepsilon'^j_p; j = 1 \text{ to } J), \quad (8)$$

where ε_p^{ij} is the precision measure of the j th line segment, defined using (5). The net reliability measure of the digital curve is defined as follows (9) [21]:

$$\varepsilon_r = \frac{\sum_{j=1}^J \sum_i |x_i^j a^j + y_i^j b^j - 1|}{\sum_{j=1}^J s_{\max}^j}, \quad (9)$$

where $s_{\max}^j$ corresponds to $s_{\max}$ defined after (6) for the j th line segment.

The previous definitions in the context of a digital curve are consistent with the concepts of precision and reliability presented in [20]. Since precision is a local property of fit, it can be defined only for the individual line segments and therefore the net precision accounts for the precision of every line segment. On the other hand, reliability is the characteristic of the complete digital curve. Thus, the net reliability measure does not consider the reliability measure of individual segment, but the individual constituents of reliability measure (the numerator and denominator in (9)) are computed using all the segments.

5. Numerical Experiments

In this section, three numerical experiments are presented. The first experiment (Section 5.3) uses 17 very small images, each with one digital curve, so that the performances of the various methods can be seen in close up for various curvature conditions of the digital curves. In the second experiment (Section 5.4), the traditional examples used in the research related to dominant point detection methods [13, 14, 16, 17, 41–43] are used for benchmarking the proposed method against Masood [14], Carmona-Poyato et al. [16], Marji and Siy [41], Teh and Chin [9], Ansari and Huang [10], B. K. Ray and K. S. Ray [12], B. K. Ray and K. S. Ray [42], Arcelli and Ramella [43], Sarkar [13], and Cronin [11]. In the third experiment (Section 5.5), 10 datasets with a total of 102628 images are considered in the study on the performance of various methods for huge datasets. The various algorithms used for comparison are described in Section 5.1 and the performance parameters used to quantitatively compare the performances of various methods in Section 5.2.

5.1. Algorithms for Comparison. For the ease of reference, the following nomenclatures are used for the line fitting methods. The method proposed by RDP [39, 40], as discussed in Section 2, is called RDP<tol>. For example, RDP2 refers to the use of tolerance value for maximum deviation of 2 pixels in the RDP algorithm. The proposed method is called RDP<number>, where <number> denotes the value of ε_0 . For example, PRO0.2 refers to the use of $\varepsilon_0 = 0.2$ in the PRO method. The method proposed by Carmona-Poyato et al. [16] is referred to as Carmona and uses 0.4 as the value of the control parameter used in [16]. The method proposed by Masood [14] is referred to as Masood and uses 0.9 as the value of its control parameter, maximum tolerable deviation of the pixels from the fitted lines. Carmona and Masood are chosen for comparison because of two reasons. The first

reason is that they are quite recent and represent the state-of-the-art methods for dominant point detection problem. The second reason is that while Masood uses local nature of fit as the main criterion for dominant point detection, Carmona-Poyato uses the global nature of fit as the main criterion. Thus, it is interesting to compare the performance of these methods against PRO, which considers both the local and global natures of fit. A total of 8 algorithms, namely, PRO0.2, PRO0.6, PRO1.0, RDP1, RDP2, RDP3, Carmona, and Masood are considered for detailed comparison in all experiments.

5.2. Performance Metrics. Following eight performance metrics are considered, where L is the number of images in a dataset, K is the number of digital curves in the l th image, and J is the number of line segments fit on the k th curve.

- (1) Maximum precision measure per line segment (MPLS): maximum of the precision measures given by (5) of all the line segments fitted for the digital curves in an image is computed. Then the quantity is averaged for all the images in the dataset and is given in (10):

$$\text{MPLS} = \text{mean} \left(\left(\varepsilon_p' \right)_{\text{Image}}^l; l = 1 \text{ to } L \right), \quad (10)$$

$$\left(\varepsilon_p' \right)_{\text{Image}} = \max \left(\left(\varepsilon_p' \right)_{\text{eq. (5)}}^k; k = 1 \text{ to } K \right).$$

- (2) Maximum reliability measure per line segment (MRLS): maximum of the reliability measures given by (6) of all the line segments fitted for the digital curves in an image is computed. Then the quantity is averaged for all the images in the dataset as given in (11)

$$\text{MRLS} = \text{mean} \left(\left(\varepsilon_r \right)_{\text{Image}}^l; l = 1 \text{ to } L \right), \quad (11)$$

$$\left(\varepsilon_r \right)_{\text{Image}} = \max \left(\left(\varepsilon_r \right)_{\text{eq. (6)}}^k; k = 1 \text{ to } K \right).$$

- (3) Maximum precision measure per digital curve (MPDC): maximum of the precision measures given by (8) for all the digital curves in an image is computed. Then the quantity is averaged for all the images in the dataset as shown in (12)

$$\text{MPDC} = \text{mean} \left(\left(\varepsilon_p' \right)_{\text{Image}}^l; l = 1 \text{ to } L \right), \quad (12)$$

$$\left(\varepsilon_p' \right)_{\text{Image}} = \max \left(\left(\varepsilon_p' \right)_{\text{eq. (8)}}^k; k = 1 \text{ to } K \right).$$

- (4) Maximum reliability measure per digital curve (MRDC): maximum of the reliability measures given by (9) for all the digital curves in an image is computed. Then the quantity is averaged for all the images in the dataset as shown in (13)

$$\text{MRDC} = \text{mean} \left(\left(\varepsilon_r \right)_{\text{Image}}^l; l = 1 \text{ to } L \right), \quad (13)$$

$$\left(\varepsilon_r \right)_{\text{Image}} = \max \left(\left(\varepsilon_r \right)_{\text{eq. (9)}}^k; k = 1 \text{ to } K \right).$$

- (5) Average dimensionality reduction (ADR): average of the dimensionality reduction measures of all the digital curves in an image is computed. Then the quantity is averaged for all the images in the dataset as shown in (14)

$$\text{ADR} = \text{mean} \left(\text{mean} \left((\text{DR})^k; k = 1 \text{ to } K \right); l = 1 \text{ to } L \right),$$

$$\text{DR} = \frac{M}{(J + 1)}, \quad (14)$$

where M is the number of pixels in a digital curve. DR is the reciprocal of the commonly known compression ratio.

- (6) Maximum deviation ($E_{\max}$): for a digital curve segment represented by a single line connecting two adjacent dominant points, the maximum deviation (distance) of the pixels on the digital curve segment from the line segment is found [18, 19, 44]. For a complete digital curve, the maximum of the maximum deviations for all the curve segments is found. This quantity is averaged for all the curves in an image and then averaged for all the images in a dataset. It is represented as $E_{\max}$ as shown in (15)

$$E_{\max} = \text{mean} \left(\text{mean} \left(E_{\max}^k; k = 1 \text{ to } K \right); l = 1 \text{ to } L \right)$$

$$E_{\max}^k = \max \left(\left(\frac{\|\mathbf{XA} - \bar{\mathbf{J}}\|_{\infty}}{|\bar{\mathbf{A}}|} \right); j = 1 \text{ to } J \right), \quad (15)$$

where $\|\cdot\|_{\infty}$ denotes the infinity norm (i.e., maximum norm).

- (7) Integral square error (ISE): for a digital curve segment represented by a single line connecting two adjacent dominant points, the sum of squares of the deviations (distances) of the pixels on the digital curve segment from the line segment is found [18, 19, 44]. For a complete digital curve, the sum of squares of the deviations for all the curve segments is added. This quantity is averaged for all the curves in an image and then averaged for all the images in a dataset. It is represented as ISE as shown in (16)

$$\text{ISE} = \text{mean} \left(\text{mean} \left(\text{ISE}_k; k = 1 \text{ to } K \right); l = 1 \text{ to } L \right)$$

$$\text{ISE}_k = \sum_{j=1}^J \left(\frac{|\mathbf{XA} - \bar{\mathbf{J}}|}{|\bar{\mathbf{A}}|} \right)^2. \quad (16)$$

- (8) Figure of merit (FOM): for a digital curve, fom is computed as the reciprocal of the product of the integral square error (ISE) and the dimensionality reduction ratio (DR) [18, 19, 44]. This quantity is averaged for all the curves in an image and then averaged for all

the images in a dataset. It is represented as FOM as shown in (17)

$$\text{FOM} = \text{mean} \left(\text{mean} \left(\text{FOM}_k; k = 1 \text{ to } K \right); l = 1 \text{ to } L \right)$$

$$\text{FOM}_k = \left((\text{DR})_{\text{eq. (10)}}^k \times \sum_{j=1}^J \left(\frac{|\mathbf{XA} - \bar{\mathbf{J}}|}{|\bar{\mathbf{A}}|} \right)^2 \right)^{-1}. \quad (17)$$

For all the previous parameters except FOM, it is desirable that their values be as close to zero as possible. In practice, they are all positive and may not be upper bounded. On the other hand, it is desirable to have high value of FOM, though it is notable that FOM is biased towards very close fitting [18, 19].

5.3. Experiment 1. In this experiment, we consider the 17 examples used in [20]. Each example is a snippet of 20 pixels dimensions containing only one digital curve, which may be similar to a digital line or a more complicated curve. The actual lines fitted by various algorithms are shown in Figure 1. Further, the performance parameters are listed in Table 1.

It can be seen in the row PRO0.2 of Figure 1 that PRO0.2 tends to follow the digital curves very closely. As a consequence it is very sensitive to the effect of digitization and generates numerous small line segments to represent the curve, strongly evident in columns (e–h) of Figure 1. Though definitely very reliable and precise, as evident from MPLS, MPRS, MPDC, and MRDC values in Table 1, due to the tendency to fit the curves very closely (note the low values of $E_{\max}$ and ISE in Table 1), they perform poorly in dimensionality reduction (see ADR values in Table 1). The FOM goes to infinity due to the exact fitting for snippets (a)–(d). So, the value of FOM for this experiment for all the algorithms has been excluded. Next, we see that PRO0.6 tends to follow the curvature of the digital curve, better than PRO0.2. We highlight the results in column (m) of Figure 1. While PRO0.2 generated many line segments for the right side of the curve, PRO0.6 is more selective in fitting the line segments and fits the line segments focusing at the location of changes in curvature, rather than following every small-scale feature of the curve. This is significantly evident in the results in columns (i), (j), and (q) of Figure 1. In PRO1.0, instead of focusing on the small features in the digital curve, it tends to follow the general characteristics of the digital curve on a relatively larger scale; see columns (m) and (n) of Figure 1. As a consequence of this characteristic, PRO1.0 has significantly better dimensionality reduction as compared to other PRO algorithms (see ADR in Table 1).

PRO algorithms demonstrate at least two well-defined patterns in terms of performance and the nature of fit. The first pattern, corresponding to tolerance values close to zero, provides very close fits that are highly reliable and precise but has poor dimensionality reduction. The second pattern, corresponding to tolerance values close to 1, performs in between RDP1 and RDP2 techniques and tends to smooth out small variations and retain large-scale curvature changes.

Next, the performance of RDP(tol) algorithms is considered. RDP1 algorithm gives a performance comparable to

FIGURE 1: Results for the 17 snippets used in [20]. The lines fitted by various algorithms are shown for each snippet.

TABLE 1: The values of performance parameters (Experiment 1) for various algorithms. The entries in bold are the minimum values of the performance parameters in a group of algorithms.

	MPLS	MRLS	MPDC	MRDC	ADR	$E_{\max}$	ISE
PRO0.2	0.02	0.02	0.00	0.00	0.39	0.03	0.02
PRO0.6	0.26	0.20	0.20	0.17	0.14	0.46	1.69
PRO1.0	0.34	0.27	0.32	0.25	0.12	0.61	3.98
RDP1	0.29	0.22	0.23	0.19	0.14	0.47	2.22
RDP2	0.41	0.32	0.36	0.29	0.12	0.69	5.56
RDP3	0.51	0.42	0.45	0.37	0.11	0.87	9.52
Masood [14]	0.49	0.41	0.30	0.31	0.18	0.88	8.89
Carmona-Poyato et al. [16, 19]	0.18	0.14	0.13	0.11	0.24	0.30	1.90

that of PRO0.6, both qualitatively (specifically note columns (i), (m), and (o) of Figure 1) and quantitatively (see Table 1). RDP2 and RDP3 perform poorer than RDP1 for all the parameters except ADR.

Now, the methods of Masood [14] and Carmona-Poyato et al. [16] are considered. In general, [16] reported that Carmona may sacrifice to some extent on the maximum deviation in order to maintain the general characteristics of the curve; the results in Figure 1 show a contradictory result. However, the results of Masood are in general consistent with the general trends observed in [14]. The anomaly of Carmona's results is explained as follows. Carmona's method attempts to fit the line segments using a relative measure such that the allowable maximum deviation increases with the length of the digital curve. This results in overfitting for small digital curves and underfitting for large digital curves. These trends are more clearly visible in Sections 5.4 and 5.5.

5.4. Experiment 2. In this experiment, we consider the digital curves considered in [13, 14, 16, 17, 41–43], namely, chromosome, leaf, semicircle, infinity, dog, maple leaf, and Africa (for dog, maple leaf, and Africa, the images from [14] were scanned with a resolution of 300 dpi, the scanned images were blurred using Adobe Photoshop with a blur radius of 2 pixels, and the blurred images were then subjected to threshold to obtain the binary images). The algorithms were executed on a notebook with Intel Core i7 CPU (M620@2.67 GHz), 4 GB RAM, 64 bit Windows 7 system using Matlab 2010. The results obtained are plotted in Figure 2. The qualitative results show that PRO0.6, PRO1.0, and RDP1 are effective in representing all the digital curves. On the other hand, PRO0.2 is very ineffective in representing the digital curves. At best, it is close to the break points discussed in [14].

On the other hand RDP2 and RDP3 are ineffective due to underfitting, especially for small curves like chromosome and infinity. The results of Masood [14] show that Masood is effective in representing the curvature well for most cases. However, for some cases, especially for larger curves, Masood tends to overfit the curves (as notable from the results of dog, maple leaf, and Africa in Figure 2 and Table 2), resulting in very small values of $E_{\max}$ and ISE and high value of ADR. On the other hand, Carmona-Poyato et al. [16] represents the large digital curves well, as noted from the low value of ADR and reasonable values of MPDC and MRDC; it has a tendency

to underfit (crude fitting) for smaller curves like chromosome and infinity. The numerical results for chromosome, leaf, and semicircle for the methods Marji and Siy [41], Teh and Chin [9], Ansari and Huang [10], B. K. Ray and K. S. Ray [12], B. K. Ray and K. S. Ray [42], Arcelli and Ramella [43], Sarkar [13], and Cronin [11] have been calculated from the dominant points reported by the methods in their respective publications.

The quantitative comparison of all the algorithms considered in this paper is listed in Table 2. Quantitative comparison gives more interesting insights. The values of MPLS, MRLS, and $E_{\max}$ are exactly the same for all the examples for PRO0.2. Interestingly, for these examples, PRO0.2 fits line segments with zero maximum deviations except for the case shown in Figure 3. This is because for these examples (Figure 2), for any other pixel sequence, over which PRO0.2 tries to fit a single line segment, either the precision measure ϵ'_p or the reliability measure ϵ_r exceeds the threshold value $\epsilon_0 = 0.2$.

Another interesting observation is that PRO0.2 always gives the best (highest) FOM. This is because the figure of merit is inversely proportional to the ISE and for PRO0.2, ISE is very small. Indeed FOM is not the best metric for quantifying the quality of line fit, as concluded in [18, 19]. It is interesting to observe that the values of MPLS, MRLS, MPDC, and MRDC have direct association with the quality of fitting in Figure 2. We note that among all the algorithms considered, Masood [14] (being highly iterative by the design of the algorithm) takes the maximum time and the time taken by Masood increases exponentially with the length of the digital curves. All the remaining algorithms have similar computation time.

The methods having similar quality of fit have similar values of these parameters, though the values of ISE and FOM may vary greatly. For example, for maple leaf compare the values of PRO1.0 and RDP2 or Carmona-Poyato et al. [16] and RDP3. This indicates that the proposed metrics MPLS, MRLS, MPDC, and MRDC are good at representing the quality of fit over various methods.

5.5. Experiment 3. In this experiment, various datasets used in diverse image processing applications like segmentation, object detection, object recognition, object categorization, shape and contour based analysis, virtual reality, and gaming applications are considered. The datasets considered are

FIGURE 2: The digital curves used in Masood [14] and Carmona-Poyato et al. [16] and the lines fitted using PRO0.2, PRO0.6, PRO1.0, RDP1, RDP2, RDP3, Masood, and Carmona.

TABLE 2: Quantitative comparison of the results of Experiment 2 for standard digital curves [14, 16]. (N : number of dominant points).

	MPLS	MRLS	MPDC	MRDC	ADR	$E_{\max}$	ISE	FOM	Time (sec)	N
Chromosome	Number of pixels = 61									
PRO0.2	0.1961	0.1538	0.0056	0.0074	0.5902	0.2774	0.1538	11.0139	0.1789	35
PRO0.6	0.5606	0.5000	0.3559	0.2821	0.1967	0.8944	11.0088	0.4618	0.0454	11
PRO1.0	0.9250	0.8200	0.4022	0.3222	0.1803	2.1213	16.3088	0.3400	0.0423	10
RDP1	0.5606	0.5000	0.3559	0.2821	0.1967	0.8944	11.0088	0.4618	0.0479	11
RDP2	1.2566	1.0032	0.7142	0.7065	0.1475	2.0000	59.9961	0.1130	0.0310	8
RDP3	1.4320	1.5800	1.1293	0.9662	0.1148	2.3016	91.1803	0.0956	0.0235	6
Masood [14]	0.5298	0.3962	0.2607	0.2210	0.1967	1.2362	7.6233	0.6668	0.2368	11
Carmona-Poyato et al. [16, 19]	0.9798	1.1068	0.4103	0.3946	0.1803	1.5855	23.7435	0.2336	0.0293	10
Marji and Siy [41]	0.5298	0.3962	0.3252	0.2650	0.1803	1.2362	10.0067	0.5542	—	11
Teh and Chin [9]	3.0075	5.7931	0.5010	0.6940	0.2295	5.3852	124.8102	0.0349	—	14
Ansari and Huang [10]	0.9487	1.1000	0.3544	0.3226	0.2787	1.5811	16.1670	0.2219	—	17
B. K. Ray and K. S. Ray [12]	0.3835	0.2941	0.2087	0.1593	0.3115	0.6860	5.1227	0.6267	—	19
B. K. Ray and K. S. Ray [42]	0.3589	0.2459	0.2171	0.1601	0.3115	0.6402	4.8897	0.6566	—	19
Arcelli and Ramella [43]	1.4320	1.5405	0.6943	0.5715	0.1311	2.3016	44.2227	0.1724	—	8
Sarkar [13]	0.3101	0.2353	0.1341	0.1511	0.3279	0.5547	3.8567	0.7908	—	20
Cronin [11]	0.3162	0.2500	0.1400	0.1081	0.2951	0.6325	3.1812	1.0653	—	18
Leaf	Number of pixels = 124									
PRO0.2	0.1961	0.1538	0.0036	0.0039	0.4516	0.2774	0.1538	14.3929	0.3427	55
PRO0.6	0.5621	0.6000	0.3257	0.2935	0.1774	1.0525	21.6953	0.2598	0.0911	21
PRO1.0	0.8165	0.7838	0.4123	0.3795	0.1532	1.3152	36.6993	0.1778	0.0712	18
RDP1	0.5621	0.6000	0.3020	0.2679	0.1855	0.9939	19.1722	0.2812	0.1013	22
RDP2	1.3651	1.1142	0.5172	0.4918	0.1371	2.0000	65.4687	0.1114	0.0598	16
RDP3	1.3651	1.1142	0.5172	0.4918	0.1371	2.0000	65.4687	0.1114	0.0598	16
Masood [14]	0.4243	0.3909	0.2223	0.1983	0.1855	0.9487	12.7075	0.4243	0.6137	22
Carmona-Poyato et al. [16, 19]	0.5164	0.4698	0.2652	0.2431	0.1694	0.9899	16.8155	0.3512	0.0681	20
Marji and Siy [41]	0.7259	0.6034	0.4056	0.3605	0.1452	1.2957	31.3875	0.2195	—	18
Teh and Chin [9]	0.5621	0.5488	0.2062	0.2339	0.2419	1.0525	18.5508	0.2228	—	30
Ansari and Huang [10]	0.7906	1.5000	0.2665	0.2666	0.2419	1.4142	24.1675	0.1710	—	30
B. K. Ray and K. S. Ray [12]	0.5648	0.5410	0.2133	0.2069	0.2581	0.9959	15.3456	0.2525	—	32
B. K. Ray and K. S. Ray [42]	0.6112	0.5410	0.2027	0.1993	0.2661	1.1142	15.0609	0.2495	—	33
Arcelli and Ramella [43]	1.3651	1.1041	0.4999	0.4968	0.1371	1.9508	64.2461	0.1135	—	17
Sarkar [13]	0.6112	0.5172	0.2835	0.2614	0.1935	1.1142	18.0333	0.2865	—	24
Cronin [11]	0.4873	0.4500	0.1700	0.1397	0.2339	0.9487	8.9810	0.4761	—	29
Semicircle	Number of pixels = 103									
PRO0.2	0.1961	0.1538	0.0178	0.0190	0.4369	0.2774	0.6154	3.7194	0.2545	44
PRO0.6	0.5566	0.4148	0.3025	0.2732	0.1553	1.5860	18.1298	0.3551	0.0715	15
PRO1.0	0.9539	0.8400	0.4547	0.3955	0.1262	1.5556	33.7812	0.2345	0.0494	12
RDP1	0.4851	0.3448	0.2604	0.2159	0.1845	1.0000	13.6577	0.3969	0.0814	18
RDP2	0.9539	0.8400	0.4547	0.3955	0.1262	1.5556	33.7812	0.2345	0.0495	12
RDP3	1.6884	1.7692	0.7132	0.5987	0.1068	2.6047	77.7992	0.1204	0.0374	10
Masood [14]	0.5086	0.4054	0.1389	0.1021	0.2621	1.1508	5.3936	0.7073	0.4405	26
Carmona-Poyato et al. [16, 19]	0.4988	0.4390	0.2255	0.2061	0.2136	0.8835	11.9412	0.3921	0.0514	21
Marji and Siy [41]	0.7259	0.6034	0.4056	0.3605	0.1452	1.2957	31.3875	0.2195	—	16
Teh and Chin [9]	0.5621	0.5488	0.2062	0.2339	0.2419	1.0525	18.5508	0.2228	—	24
Ansari and Huang [10]	0.7906	1.5000	0.2665	0.2666	0.2419	1.4142	24.1675	0.1710	—	29
B. K. Ray and K. S. Ray [12]	0.5648	0.5410	0.2133	0.2069	0.2581	0.9959	15.3456	0.2525	—	30
B. K. Ray and K. S. Ray [42]	0.6112	0.5410	0.2027	0.1993	0.2661	1.1142	15.0609	0.2495	—	28
Arcelli and Ramella [43]	1.3651	1.1041	0.4999	0.4968	0.1371	1.9508	64.2461	0.1135	—	10
Sarkar [13]	0.6112	0.5172	0.2835	0.2614	0.1935	1.1142	18.0333	0.2865	—	20
Cronin [11]	0.4873	0.4500	0.1700	0.1397	0.2339	0.9487	8.9810	0.4761	—	31

TABLE 2: Continued.

	MPLS	MRLS	MPDC	MRDC	ADR	$E_{\max}$	ISE	FOM	Time (sec)	N
Infinity										
	Number of pixels = 46									
PRO0.2	0.1961	0.1538	0.0070	0.0101	0.6304	0.2774	0.1538	10.3103	0.1532	28
PRO0.6	0.5684	0.5385	0.3128	0.2341	0.2174	1.1094	6.2977	0.7304	0.0329	9
PRO1.0	0.8793	0.7123	0.4830	0.4264	0.1739	1.6386	19.9482	0.2882	0.0222	7
RDP1	0.4758	0.3585	0.3042	0.2458	0.2174	0.9713	6.6735	0.6893	0.0363	9
RDP2	0.8793	0.7123	0.4830	0.4264	0.1739	1.6386	19.9482	0.2882	0.0225	7
RDP3	1.5479	1.5479	0.9760	0.8147	0.1304	2.3408	53.8220	0.1428	0.0195	5
Masood [14]	0.7268	0.5846	0.3556	0.3096	0.1957	1.4033	10.2482	0.4987	0.1503	8
Carmona-Poyato et al. [16, 19]	1.1243	0.9600	0.6911	0.5871	0.1522	1.8000	31.6866	0.2074	0.0266	6
Dog										
	Number of pixels = 458									
PRO0.2	0.1961	0.1538	0.0055	0.0062	0.4672	0.2774	0.9231	2.3185	1.6761	213
PRO0.6	0.5943	0.5098	0.2748	0.2467	0.1288	1.4486	73.6651	0.1054	0.3485	58
PRO1.0	0.9841	0.8714	0.5626	0.4635	0.0873	2.1374	207.8683	0.0551	0.2178	39
RDP1	0.8321	0.7500	0.2829	0.2397	0.1332	1.0000	71.7364	0.1047	0.3689	60
RDP2	1.2774	1.1647	0.5298	0.4341	0.0917	1.9524	190.9351	0.0571	0.2283	41
RDP3	1.8932	1.6500	0.8170	0.6867	0.0764	2.8460	454.1739	0.0288	0.2083	34
Masood [14]	0.3780	0.3000	0.1203	0.1155	0.2249	1.0000	21.3056	0.2087	26.4447	102
Carmona-Poyato et al. [16, 19]	0.9238	0.7844	0.3349	0.3152	0.1157	1.7171	115.3301	0.0749	0.6452	52
Maple leaf										
	Number of pixels = 504									
PRO0.2	0.1961	0.1538	0.0094	0.0113	0.5000	0.2774	1.8462	1.0833	2.1433	251
PRO0.6	0.5774	0.5000	0.2815	0.2379	0.1250	1.2308	69.1412	0.1157	0.4042	62
PRO1.0	0.9623	0.7056	0.4776	0.4043	0.0853	2.0000	191.9178	0.0611	0.2627	42
RDP1	0.5606	0.4667	0.2543	0.2176	0.1329	1.0000	58.8400	0.1278	0.4355	66
RDP2	1.0871	0.9542	0.5028	0.4467	0.0813	2.0000	218.7481	0.0562	0.2436	40
RDP3	1.6786	1.5244	1.0533	0.8735	0.0456	2.8943	724.6112	0.0302	0.1356	22
Masood [14]	0.7071	0.6667	0.2003	0.1742	0.1389	1.0000	36.7885	0.1957	52.6449	69
Carmona-Poyato et al. [16, 19]	1.9524	1.6100	1.1309	0.9602	0.0437	3.3661	871.9568	0.0263	0.7954	21
Africa										
	Number of pixels = 431									
PRO0.2	0.1961	0.1538	0.0123	0.0132	0.4478	0.2774	1.8462	1.2096	1.4901	192
PRO0.6	0.5572	0.4595	0.3255	0.2927	0.0905	1.3868	83.4013	0.1325	0.2132	38
PRO1.0	0.9746	0.8985	0.5518	0.4355	0.0603	2.0000	175.1830	0.0946	0.1322	25
RDP1	0.6417	0.5000	0.2755	0.2450	0.1114	1.0000	61.3082	0.1465	0.2818	47
RDP2	1.0024	0.8985	0.5930	0.4758	0.0580	1.9723	203.4529	0.0847	0.1246	24
RDP3	1.4404	1.2574	0.6865	0.5469	0.0534	2.4244	274.5463	0.0683	0.1185	22
Masood [14]	0.4880	0.3814	0.2147	0.1909	0.1206	1.0153	36.3878	0.2278	26.0222	51
Carmona-Poyato et al. [16, 19]	0.9354	0.7803	0.4861	0.4023	0.0603	2.0000	148.2542	0.1118	0.5915	25

FIGURE 3: The only pixel sequence for which PRO0.2 fits a single line.

TABLE 3: Performance of various methods for all the datasets for Experiment 3.

(a) Maximum precision measure per line segment (MPLS)

	Afreight	Google	Berkley	Cars_INRIA	Caltech 101	Caltech 256	PASCAL 2007	PASCAL 2008	PASCAL 2009	PASCAL 2010	Mean
PRO0.2	0.19	0.19	0.20	0.20	0.20	0.20	0.20	0.20	0.20	0.20	0.20
PRO0.6	0.59	0.59	0.60	0.60	0.60	0.60	0.60	0.60	0.60	0.60	0.60
PRO1.0	0.97	0.97	0.99	1.00	0.99	0.99	1.00	1.00	1.00	1.00	0.99
RDP1	0.77	0.75	0.84	0.90	0.83	0.82	0.87	0.87	0.87	0.87	0.84
RDP2	1.32	1.33	1.44	1.54	1.43	1.43	1.51	1.51	1.51	1.50	1.45
RDP3	1.89	1.95	2.06	2.19	2.06	2.08	2.14	2.14	2.15	2.13	2.08
Masood [14]	1.25	1.16	1.63	1.78	1.47	1.51	1.72	1.71	1.73	1.72	1.57
Carmona-Poyato et al. [16, 19]	1.19	2.44	1.58	2.07	1.90	2.35	2.17	2.18	2.15	2.19	2.02

(b) Maximum reliability measure per line segment (MRLS)

	Afreight	Google	Berkley	Cars_INRIA	Caltech 101	Caltech 256	PASCAL 2007	PASCAL 2008	PASCAL 2009	PASCAL 2010	Mean
PRO0.2	0.15	0.15	0.15	0.15	0.15	0.15	0.15	0.15	0.15	0.15	0.15
PRO0.6	0.51	0.51	0.56	0.58	0.55	0.55	0.57	0.57	0.57	0.57	0.55
PRO1.0	0.90	0.89	0.96	0.99	0.96	0.96	0.98	0.98	0.98	0.98	0.96
RDP1	0.72	0.69	0.82	0.88	0.79	0.78	0.85	0.85	0.85	0.84	0.81
RDP2	1.49	1.40	1.74	1.98	1.68	1.66	1.86	1.84	1.86	1.83	1.73
RDP3	2.53	2.32	3.11	4.02	3.08	3.06	3.53	3.52	3.52	3.51	3.22
Masood [14]	1.23	1.10	1.73	1.93	1.51	1.52	1.79	1.79	1.80	1.80	1.62
Carmona-Poyato et al. [16, 19]	1.12	2.16	1.47	1.99	1.74	2.13	2.03	2.02	2.00	2.05	1.87

(c) Maximum precision measure per digital curve (MPDC)

	Afreight	Google	Berkley	Cars_INRIA	Caltech 101	Caltech 256	PASCAL 2007	PASCAL 2008	PASCAL 2009	PASCAL 2010	Mean
PRO0.2	0.01	0.01	0.00	0.00	0.01	0.01	0.01	0.01	0.01	0.01	0.01
PRO0.6	0.30	0.28	0.11	0.31	0.28	0.28	0.28	0.28	0.27	0.27	0.27
PRO1.0	0.44	0.43	0.21	0.49	0.43	0.44	0.43	0.42	0.41	0.42	0.41
RDP1	0.29	0.28	0.11	0.32	0.28	0.28	0.28	0.27	0.27	0.26	0.26
RDP2	0.51	0.53	0.26	0.54	0.52	0.48	0.50	0.48	0.48	0.49	0.48
RDP3	0.73	0.78	0.32	0.69	0.63	0.67	0.64	0.63	0.63	0.61	0.63
Masood [14]	0.89	0.83	1.17	1.34	1.11	1.13	1.30	1.29	1.29	1.29	1.16
Carmona-Poyato et al. [16, 19]	0.87	1.18	1.19	1.49	1.27	1.41	1.49	1.48	1.48	1.49	1.34

(d) Maximum reliability measure per digital curve (MRDC)

	Afreight	Google	Berkley	Cars_INRIA	Caltech 101	Caltech 256	PASCAL 2007	PASCAL 2008	PASCAL 2009	PASCAL 2010	Mean
PRO0.2	0.01	0.01	0.00	0.01	0.01	0.01	0.01	0.01	0.01	0.01	0.01
PRO0.6	0.26	0.23	0.09	0.27	0.23	0.23	0.24	0.23	0.23	0.23	0.22
PRO1.0	0.37	0.37	0.17	0.41	0.37	0.36	0.36	0.35	0.35	0.35	0.34
RDP1	0.25	0.23	0.09	0.27	0.24	0.24	0.24	0.23	0.23	0.23	0.23
RDP2	0.44	0.46	0.22	0.47	0.46	0.40	0.44	0.42	0.42	0.43	0.42
RDP3	0.69	0.71	0.31	0.63	0.59	0.61	0.59	0.58	0.60	0.56	0.59
Masood [14]	0.86	0.76	1.26	1.42	1.12	1.12	1.32	1.33	1.32	1.32	1.18
Carmona-Poyato et al. [16, 19]	0.84	1.23	1.13	1.58	1.23	1.44	1.50	1.52	1.50	1.53	1.35

(e) Average dimensionality reduction (ADR)

	Afreight	Google	Berkley	Cars_INRIA	Caltech 101	Caltech 256	PASCAL 2007	PASCAL 2008	PASCAL 2009	PASCAL 2010	Mean
PRO0.2	0.40	0.39	0.47	0.45	0.48	0.47	0.38	0.40	0.38	0.39	0.42
PRO0.6	0.16	0.13	0.21	0.20	0.18	0.17	0.11	0.11	0.11	0.11	0.15
PRO1.0	0.13	0.10	0.17	0.16	0.14	0.13	0.09	0.09	0.08	0.09	0.12
RDP1	0.16	0.14	0.21	0.20	0.19	0.17	0.12	0.12	0.12	0.12	0.16
RDP2	0.12	0.10	0.16	0.15	0.14	0.13	0.08	0.08	0.08	0.08	0.11
RDP3	0.10	0.08	0.13	0.12	0.11	0.10	0.07	0.07	0.07	0.07	0.09
Masood [14]	0.16	0.13	0.19	0.19	0.18	0.16	0.17	0.17	0.17	0.18	0.17
Carmona-Poyato et al. [16, 19]	0.22	0.13	0.24	0.24	0.21	0.19	0.21	0.21	0.21	0.21	0.21

(f) Maximum deviation ($E_{\max}$)

	Afreight	Google	Berkley	Cars_INRIA	Caltech 101	Caltech 256	PASCAL 2007	PASCAL 2008	PASCAL 2009	PASCAL 2010	Mean
PRO0.2	0.35	0.38	0.44	0.74	0.43	0.47	0.59	0.58	0.58	0.56	0.51
PRO0.6	1.33	1.33	1.46	1.69	1.49	1.50	1.64	1.63	1.64	1.63	1.53
PRO1.0	2.06	2.10	2.31	2.65	2.32	2.34	2.53	2.54	2.55	2.54	2.39
RDP1	1.00	0.99	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00	1.00
RDP2	1.97	1.96	2.00	2.00	2.00	1.99	2.00	2.00	2.00	2.00	1.99
RDP3	2.90	2.92	2.99	3.00	2.99	2.99	3.00	3.00	2.99	3.00	2.98
Masood [14]	2.22	2.05	0.94	2.32	2.62	2.75	3.08	3.06	3.11	3.08	2.52
Carmona-Poyato et al. [16, 19]	2.05	3.91	2.63	3.49	3.17	3.83	3.59	3.63	3.57	3.63	3.35

(g) Integral square error (ISE)

	Afreight	Google	Berkley	Cars_INRIA	Caltech 101	Caltech 256	PASCAL 2007	PASCAL 2008	PASCAL 2009	PASCAL 2010	Mean
PRO0.2	0.57	2.02	0.91	1.77	1.33	1.63	1.50	1.53	1.62	1.52	1.44
PRO0.6	26.81	104.08	33.95	79.63	58.44	69.66	64.36	64.15	66.35	61.50	62.89
PRO1.0	59.15	223.97	72.15	180.79	128.52	155.97	146.07	145.18	146.00	139.92	139.77
RDP1	27.03	95.15	32.90	69.67	53.39	62.14	57.64	59.48	60.48	54.44	57.23
RDP2	72.44	301.13	96.81	212.56	161.30	192.01	178.20	183.98	181.87	166.89	174.72
RDP3	154.28	739.04	200.78	480.41	356.05	470.48	398.36	406.30	442.71	387.20	403.56
Masood [14]	39.77	64.52	72.68	81.58	54.97	64.17	83.32	82.04	86.37	77.45	70.69
Carmona-Poyato et al. [16, 19]	71.73	5171.01	149.44	754.50	417.71	1690.91	639.90	617.49	600.87	728.07	1084.16

(h) Figure of merit (FOM)

	Afreight	Google	Berkley	Cars_INRIA	Caltech 101	Caltech 256	PASCAL 2007	PASCAL 2008	PASCAL 2009	PASCAL 2010	Mean
PRO0.2	∞	∞	3.65	1.94	2.50	2.18	2.12	2.13	2.03	2.14	2.33 (excluding ∞)
PRO0.6	0.32	0.24	0.20	0.09	0.14	0.14	0.12	0.12	0.12	0.13	0.16
PRO1.0	0.19	0.13	0.11	0.05	0.08	0.07	0.07	0.07	0.07	0.07	0.09
RDP1	0.32	0.24	0.20	0.09	0.16	0.15	0.13	0.13	0.13	0.14	0.17
RDP2	0.17	0.10	0.09	0.04	0.07	0.06	0.06	0.06	0.06	0.06	0.08
RDP3	0.09	0.05	0.05	0.02	0.04	0.03	0.03	0.03	0.03	0.03	0.04
Masood [14]	0.28	0.24	0.13	0.10	0.15	0.15	0.11	0.11	0.11	0.11	0.15
Carmona-Poyato et al. [16, 19]	0.15	0.07	0.05	0.02	0.03	0.02	0.02	0.02	0.02	0.02	0.04

(i) Average time taken per image

	Afreight	Google	Berkley	Cars_INRIA	Caltech 101	Caltech 256	PASCAL 2007	PASCAL 2008	PASCAL 2009	PASCAL 2010	Mean
PRO0.2	1.75	7.94	2.89	4.29	7.59	15.38	15.67	15.62	16.35	15.84	10.33
PRO0.6	0.35	0.79	0.61	0.83	1.56	2.66	3.16	3.14	3.29	3.21	1.96
PRO1.0	0.23	0.55	0.50	0.62	1.01	1.74	2.05	2.04	2.13	2.08	1.29
RDP1	0.38	0.89	2.71	1.13	1.72	2.93	3.45	3.44	3.59	3.51	2.37
RDP2	0.20	0.50	0.89	0.57	0.91	1.56	1.84	1.83	1.91	1.86	1.21
RDP3	0.16	0.40	0.60	0.37	0.66	0.94	1.36	1.39	1.45	1.44	0.88
Masood [14]	1.97	1065.14	3.03	21.38	26.69	113.75	27.24	25.88	28.93	25.64	133.96
Carmona-Poyato et al. [16, 19]	0.31	6.49	0.50	2.06	1.48	2.75	2.92	2.89	3.01	2.70	2.51

Afreight dataset [24] (920 images), Google dataset (this dataset has been formed by selecting images of various sizes from a random Google search). In this dataset the minimum dimension (length or breadth) of the images is $25n$, $n = 1$ to 100 and 50 images have been selected for each value of n . Thus, the smallest image is of size 25 pixels (length- or breadthwise or both), while the largest image is of size 2500 pixels (length- or breadthwise or both) (5000 images), Berkley dataset [27] (300 images), Cars_INRIA dataset [28] (150 images), Caltech 101 dataset [25] (9149 images), Caltech 256 dataset [26] (30608 images), PASCAL 2007 dataset [29] (9963 images), PASCAL 2008 dataset [30] (10057 images), PASCAL 2009 dataset [31] (14743 images), and PASCAL 2010 dataset [32] (21738 images). The binary edge map of the image is passed to the various algorithms after canny edge operation with threshold ($T_L = 0.1$ and $T_H = 0.2$). For all these datasets the performance parameters are listed in Table 3.

RDP3, Masood [14], and Carmona-Poyato et al. [16] have high values of MPLS and MRLS, whereas RDP3 performs the worst in terms of MPDC and MRDC. PRO0.2 always has the lowest values of MPLS, MRLS, MPDC, MRDC, $E_{\max}$, and ISE and the highest values of ADR and FOM. This indicates that PRO0.2 has a consistent characteristic of very close fitting. RDP3 has very high values of MPLS and MRLS but reasonably low values for MPDC and MRDC. It performs the best in terms of ADR but poorest among all for FOM (equaled with Carmona-Poyato et al. [16]). Since Masood [14] primarily focuses on the local nature of the fit, as compared to Carmona which concentrates on the global nature of fit, Masood has significantly low values of $E_{\max}$, low value of ISE, and higher value of FOM. However, qualitatively, the difference may not be significant. The performance of Masood and Carmona is similar in terms of MPLS, MRLS, MPDC, and MRDC. This is because the precision and reliability metrics are better designed to balance the global and local natures of fit. Masood takes significantly longer time as compared to any other method and time complexity increases with increase in the size of the image (curves) as seen from Google dataset result in Table 3(i). Finally, the quantitative data of PRO1.0 indicates that PRO1.0 avoids both extremes for all the performance parameters and provides a good balance in terms of the conflicting requirements of dominant point detection methods.

6. Conclusion

The paper has demonstrated that the PRO method, that uses precision and reliability measures in the optimization goals, can be easily tailored by changing the tolerance value. The fit is very close for small tolerance values and the closeness of the fit decreases as the value becomes close to 1. It is shown through extensive comparison that PRO0.2 chooses the dominant points such that each and every change in the curvature is retained though the number of points needed is large. On the other hand, PRO1.0 avoids the problems of underfitting as well as overfitting and provides a good balance for all the performance parameters that indicate local nature of fit, global nature of fit, dimensionality reduction, and the computation time.

As a final note, the results of dominant point detection methods for 10 practical datasets used in high-end computer vision applications are shown. It is demonstrated that PRO1.0 provides good results over all the datasets and across images of various types and sizes. Such a study is important for the research community that uses dominant point detection as one of the fundamental preprocessing steps in high-end applications [45, 46]. The proposed precision and reliability measures are effective in representing the quality of fit across various methods.

References

- [1] R. Yang and Z. Zhang, "Eye gaze correction with stereovision for video-teleconferencing," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 26, no. 7, pp. 956–960, 2004.
- [2] A. Kolesnikov and P. Fränti, "Data reduction of large vector graphics," *Pattern Recognition*, vol. 38, no. 3, pp. 381–394, 2005.
- [3] D. Brunner and P. Soille, "Iterative area filtering of multichannel images," *Image and Vision Computing*, vol. 25, no. 8, pp. 1352–1364, 2007.
- [4] S. Ozen, A. Bouganis, and M. Shanahan, "A fast evaluation criterion for the recognition of occluded shapes," *Robotics and Autonomous Systems*, vol. 55, no. 9, pp. 741–749, 2007.
- [5] A. Orzan, A. Bousseau, H. Winnemöller, P. Barla, J. Thollot, and D. Salesin, "Diffusion curves: a vector representation for smooth-shaded images," *ACM Transactions on Graphics*, vol. 27, no. 3, article 92, 2008.
- [6] J. L. G. Balboa and F. J. A. López, "Sinuosity pattern recognition of road features for segmentation purposes in cartographic

- generalization,” *Pattern Recognition*, vol. 42, no. 9, pp. 2150–2159, 2009.
- [7] G. Erus and N. Loménie, “How to involve structural modeling for cartographic object recognition tasks in high-resolution satellite images?” *Pattern Recognition Letters*, vol. 31, no. 10, pp. 1109–1119, 2010.
 - [8] A. Faure, L. Buzer, and F. Feschet, “Tangential cover for thick digital curves,” *Pattern Recognition*, vol. 42, no. 10, pp. 2279–2287, 2009.
 - [9] C.-H. Teh and R. T. Chin, “On the detection of dominant points on digital curves,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 11, no. 8, pp. 859–872, 1989.
 - [10] N. Ansari and K. W. Huang, “Non-parametric dominant point detection,” *Pattern Recognition*, vol. 24, no. 9, pp. 849–862, 1991.
 - [11] T. M. Cronin, “A boundary concavity code to support dominant point detection,” *Pattern Recognition Letters*, vol. 20, no. 6, pp. 617–634, 1999.
 - [12] B. K. Ray and K. S. Ray, “Detection of significant points and polygonal approximation of digitized curves,” *Pattern Recognition Letters*, vol. 13, no. 6, pp. 443–452, 1992.
 - [13] D. Sarkar, “A simple algorithm for detection of significant vertices for polygonal approximation of chain-coded curves,” *Pattern Recognition Letters*, vol. 14, no. 12, pp. 959–964, 1993.
 - [14] A. Masood, “Dominant point detection by reverse polygonization of digital curves,” *Image and Vision Computing*, vol. 26, no. 5, pp. 702–715, 2008.
 - [15] D. K. Prasad, C. Quek, and M. K. H. Leung, “A non-heuristic dominant point detection based on suppression of break points,” in *Image Analysis and Recognition*, A. Campilho and M. Kamel, Eds., vol. 7324, pp. 269–276, Springer, Berlin, Germany, 2012.
 - [16] A. Carmona-Poyato, F. J. Madrid-Cuevas, R. Medina-Carnicer, and R. Muñoz-Salinas, “Polygonal approximation of digital planar curves through break point suppression,” *Pattern Recognition*, vol. 43, no. 1, pp. 14–25, 2010.
 - [17] T. P. Nguyen and I. Debled-Rennesson, “A discrete geometry approach for dominant point detection,” *Pattern Recognition*, vol. 44, no. 1, pp. 32–44, 2011.
 - [18] P. L. Rosin, “Techniques for assessing polygonal approximations of curves,” *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 19, no. 6, pp. 659–666, 1997.
 - [19] A. Carmona-Poyato, R. Medina-Carnicer, F. J. Madrid-Cuevas, R. Muñoz-Salinas, and N. L. Fernández-García, “A new measurement for assessing polygonal approximation of curves,” *Pattern Recognition*, vol. 44, no. 1, pp. 45–54, 2011.
 - [20] D. K. Prasad and M. K. H. Leung, “Reliability/precision uncertainty in shape fitting problems,” in *Proceedings of the 17th IEEE International Conference on Image Processing (ICIP '10)*, pp. 4277–4280, Hong Kong, China, September 2010.
 - [21] D. K. Prasad and M. K. H. Leung, “Polygonal representation of digital curves,” in *Digital Image Processing*, S. G. Stanciu, Ed., pp. 71–90, InTech, Rijeka, Croatia, 2012.
 - [22] D. K. Prasad, *Geometric primitive feature extraction-concepts, algorithms, and applications [Ph.D. thesis]*, School of Computer Engineering, Nanyang Technological University, Singapore, 2012.
 - [23] O. Strauss, “Reducing the precision/uncertainty duality in the Hough transform,” in *Proceedings of the IEEE International Conference on Image Processing (ICIP '96)*, pp. 967–970, Lausanne, Switzerland, September 1996.
 - [24] G. McCarter and A. Storkey, *Air Freight Image Sequences*, 2003.
 - [25] L. Fei-Fei, R. Fergus, and P. Perona, “Learning generative visual models from few training examples: an incremental Bayesian approach tested on 101 object categories,” *Computer Vision and Image Understanding*, vol. 106, no. 1, pp. 59–70, 2007.
 - [26] California Institute of Technology, <http://authors.library.caltech.edu/7694>.
 - [27] D. Martin, C. Fowlkes, D. Tal, and J. Malik, “A database of human segmented natural images and its application to evaluating segmentation algorithms and measuring ecological statistics,” in *Proceedings of the 8th International Conference on Computer Vision*, pp. 416–423, July 2001.
 - [28] P. Carbonetto, G. Dorkó, C. Schmid, H. Kück, and N. de Freitas, “Learning to recognize objects with little supervision,” *International Journal of Computer Vision*, vol. 77, no. 1–3, pp. 219–237, 2008.
 - [29] M. Everingham, L. V. Gool, C. K. I. Williams, J. Winn, and A. Zisserman, *The PASCAL Visual Object Classes Challenge 2007 (VOC2007)*, 2007.
 - [30] M. Everingham, L. V. Gool, C. K. I. Williams, J. Winn, and A. Zisserman, *The PASCAL Visual Object Classes Challenge 2008 (VOC2008)*, 2008.
 - [31] M. Everingham, L. V. Gool, C. K. I. Williams, J. Winn, and A. Zisserman, *The PASCAL Visual Object Classes Challenge 2009 (VOC2009)*, 2009.
 - [32] M. Everingham, L. V. Gool, C. K. I. Williams, J. Winn, and A. Zisserman, *The PASCAL Visual Object Classes Challenge 2010 (VOC2010)*, 2010.
 - [33] D. K. Prasad, “Fabrication imperfection analysis and statistics generation using precision and reliability optimization method,” *Optics Express*, vol. 21, pp. 17602–17614, 2013.
 - [34] D. K. Prasad and M. S. Brown, “Online tracking of deformable objects under occlusion using dominant points,” *Journal of the Optical Society of America*, vol. 30, pp. 1484–1491, 2013.
 - [35] D. K. Prasad, M. K. H. Leung, C. Quek, and S.-Y. Cho, “A novel framework for making dominant point detection methods non-parametric,” *Image and Vision Computing*, vol. 30, pp. 843–859, 2012.
 - [36] D. K. Prasad, “Assessing error bound for dominant point detection,” *International Journal of Image Processing*, vol. 6, pp. 326–333, 2012.
 - [37] D. K. Prasad, C. Quek, M. K. H. Leung, and S.-Y. Cho, “A parameter independent line fitting method,” in *Proceedings of the Asian Conference on Pattern Recognition (ACPR '11)*, pp. 441–445, 2011.
 - [38] D. G. Lowe, “Three-dimensional object recognition from single two-dimensional images,” *Artificial Intelligence*, vol. 31, no. 3, pp. 355–395, 1987.
 - [39] D. H. Douglas and T. K. Peucker, “Algorithms for the reduction of the number of points required to represent a digitized line or its caricature,” *Cartographica*, vol. 10, pp. 112–122, 1973.
 - [40] U. Ramer, “An iterative procedure for the polygonal approximation of plane curves,” *Computer Graphics and Image Processing*, vol. 1, no. 3, pp. 244–256, 1972.
 - [41] M. Marji and P. Siy, “Polygonal representation of digital planar curves through dominant point detection—a nonparametric algorithm,” *Pattern Recognition*, vol. 37, no. 11, pp. 2113–2130, 2004.
 - [42] B. K. Ray and K. S. Ray, “An algorithm for detection of dominant points and polygonal approximation of digitized curves,” *Pattern Recognition Letters*, vol. 13, no. 12, pp. 849–856, 1992.
 - [43] C. Arcelli and G. Ramella, “Finding contour-based abstractions of planar patterns,” *Pattern Recognition*, vol. 26, no. 10, pp. 1563–1577, 1993.

- [44] P. L. Rosin, "Assessing the behaviour of polygonal approximation algorithms," *Pattern Recognition*, vol. 36, no. 2, pp. 505–518, 2003.
- [45] D. K. Prasad and M. K. H. Leung, "A hybrid approach for ellipse detection in real images," in *2nd International Conference on Digital Image Processing*, vol. 7546 of *Proceedings of SPIE*, p. 75460I, Singapore, February 2010.
- [46] D. K. Prasad, "Adaptive traffic signal control system with cloud computing based online learning," in *Proceedings of the 8th International Conference on Information, Communications and Signal Processing (ICICS '11)*, pp. 1–5, Singapore, December 2011.